



Docket No.: 65164-P006C1C2C1-10606085  
(PATENT)

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Patent Application of:  
Sandro Pasquali

Application No.: 10/672,945

Confirmation No.: 1566

Filed: September 29, 2003

Art Unit: 2178

For: **SYSTEM AND METHOD FOR  
FACILITATING A WINDOWS BASED  
CONTENT MANIFESTATION  
ENVIRONMENT WITHIN A WWW  
BROWSER**

Examiner: PAULA, CESAR B

**APPEAL BRIEF**

MS Appeal Brief - Patents  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Dear Sir:

As required under § 41.37(a), this brief is filed more than two months after the Notice of Appeal filed in this case on June 8, 2007, and is in furtherance of said Notice of Appeal.

The fees required under § 41.20(b)(2) are dealt with in the accompanying  
TRANSMITTAL OF APPEAL BRIEF.

This brief contains items under the following headings as required by 37 C.F.R. § 41.37 and M.P.E.P. § 1205.2.

**TABLE OF CONTENTS**

01/09/2008 MBELETE1 00000061 10672945

02 FC:2402

255.00 0P

**I. REAL PARTY IN INTEREST ..... 1**

<b>II.</b>	<b>RELATED APPEALS AND INTERFERENCES .....</b>	<b>2</b>
<b>III.</b>	<b>STATUS OF THE CLAIMS .....</b>	<b>3</b>
<b>IV.</b>	<b>STATUS OF AMENDMENTS .....</b>	<b>4</b>
<b>V.</b>	<b>SUMMARY OF CLAIMED SUBJECT MATTER.....</b>	<b>5</b>
<b>VI.</b>	<b>GROUND OF REJECTION TO BE REVIEWED ON APPEAL .....</b>	<b>8</b>
<b>VII.</b>	<b>ARGUMENT .....</b>	<b>9</b>
A.	Rejections Under 35 U.S.C. § 102(b) In View of <i>Lemay et al.</i> ....	9
1.	Claims 1, 14, 19, 30 and 40 .....	15
i.	Claim 1 .....	15
ii.	Claim 14.....	16
iii.	Claim 19.....	16
iv.	Claim 30.....	16
v.	Claim 40.....	17
2.	Dependent Claims 2, 20 and 31 are Separately Patentable.....	17
3.	Dependent Claims 3, 21 and 32 are Separately Patentable.....	18
4.	Dependent Claims 4, 22 and 33 are Separately Patentable.....	18
5.	Dependent Claims 5, 23 and 34 are Separately Patentable.....	19
6.	Dependent Claims 6, 24 and 35 are Separately Patentable.....	19
7.	Dependent Claims 7, 25 and 36 are Separately Patentable.....	20
8.	Dependent Claims 8, 26 and 37 are Separately Patentable.....	20
9.	Dependent Claims 9, 27 and 38 are Separately Patentable.....	21
10.	Dependent Claims 10, 28 and 39 are Separately Patentable.....	21
11.	Dependent Claims 11, 18, 29 and 47 are Separately Patentable.....	22
12.	Dependent Claim 12 is Separately Patentable .....	22
13.	Dependent Claim 13 is Separately Patentable .....	23
14.	Dependent Claim 15 is Separately Patentable .....	23
16.	Dependent Claim 16 is Separately Patentable .....	24
B.	Conclusion .....	24
<b>VIII.</b>	<b>CLAIMS APPENDIX.....</b>	<b>26</b>
<b>IX.</b>	<b>EVIDENCE APPENDIX.....</b>	<b>34</b>
<b>X.</b>	<b>RELATED PROCEEDINGS APPENDIX .....</b>	<b>35</b>



**I. REAL PARTY IN INTEREST**

The real party in interest for this appeal is:

Simple.com, Inc. with offices located at 77 Mowat Ave., Suite 414; Toronto, Ontario,  
Canada M6K 3E3.

## **II. RELATED APPEALS AND INTERFERENCES**

Patents in the chain of priority of the subject patent application including U.S. Patent Nos. 6,272,493; 6,434,563 and 6,535,882 are involved in litigation currently pending in the United States District Court for the Eastern District of New York entitled *Computer Associates International, Inc. v. Simple.Com, Inc., and Wired Solutions, L.L.C.*, Civil File No. 2:02-cv-02748 which proceedings have been previously filed and made of record in this case.

### **III. STATUS OF CLAIMS**

#### **A. Total Number of Claims in Application**

There are 47 claims pending in application.

#### **B. Current Status of Claims**

1. Claims canceled: None
2. Claims withdrawn from consideration but not canceled: None
3. Claims pending: 1 – 47.
4. Claims allowed: None
5. Claims rejected: 1 - 47

#### **C. Claims On Appeal**

The claims on appeal are claims 1- 47.

#### **IV. STATUS OF AMENDMENTS**

Applicant did not file an Amendment After Final Rejection. However, submitted herewith is a Terminal Disclaimer in response to all outstanding nonstatutory obviousness-type double patenting rejections.

## V. SUMMARY OF CLAIMED SUBJECT MATTER

A summary of the claimed subject matter with cross-reference to elements of the preferred embodiments described in the specification is provided below. Such cross-reference is not a representation by applicant that the scope of the claimed subject matter be limited to the preferred embodiments.

According to claim 1, a system (Fig. 1A) for facilitating a windowed content manifestation environment within a web browser comprises a server system (Fig. 1B, 102; specification at page 14, lines 3 – 17) configured to transmit a software system and associated content (Fig. 3, step S3-2) via an electronic data network (specification at page 13, lines 7 – 18; page 20, lines 26 – 31; page 26, lines 31 – 42; page 31, lines 16 – 20; page 33, lines 24 – 28; and page 56, lines 1 - 6); and a web browser client (Fig. 1C; specification at page 10, lines 14 – 17) operating within a data processing system (108) that is coupled to said server system (102) via the electronic data network (Fig. 1A) and having a content manifestation environment (Figs. 2A and 2B; specification at page 7, lines 22 – 24; page 18, lines 9 – 11; page 19, lines 12 – 21; and page 20, line 15 *et seq.*), said web browser client operative to receive said software system and said associated content via said server system (specification at page 15, lines 15 – 18; page 21, lines 4 – 8; page 30, line 45 – page 31, line 4; Fig. 5A, step S5-5; page 56, lines 26 – 31) to process said software system and said associated content to produce at least one window object (Fig. 1D; Fig. 5A, step S5-6; and page 57, lines 1 - 8) within said content manifestation environment (Fig. 3, steps S3-3 and S3-4), said at least one window object associated with a set of controllable attributes (specification at page 23, lines 7 – 21; and page 48, lines 52 – 56) and configured to manifest at least a portion of said associated content therein (Fig. 2A, page 18, lines 8 - 11), said controllable attributes configured to affect manifestation of said at least one window object by said web browser client within said content manifestation environment (Fig. 5B, step S5 – 7; specification at page 18, lines 15 – 22; and page 57, lines 10 - 15).

According to claim 14, a network client configured to operate within a data-processing system and to receive content from a remote server system to facilitate a windowed content manifestation environment comprises a content retrieval module (specification at page 15, lines 11 – 14) configured to receive content from a network server system via an electronic data network (specification at page 15, lines 2 – 11); and a processing engine coupled to said content

retrieval module (specification at page 15, lines 11 – 14) configured to operate a content manifestation environment within the data processing system (Figs. 2A and 2B; specification at page 7, lines 22 – 24; page 18, lines 9 – 11; page 19, lines 12 – 21; and page 20, line 15 *et seq.*), to process said content to produce at least one window object (Figs. 2A and 2B, page 14, lines 22 – 26) within said content manifestation environment (Fig. 1D, Fig. 2A, page 18, lines 8 – 15, Fig. 5A, step S5-6; and page 57 lines 1 -8), said at least one window object configured to manifest at least a portion of said content therein (Fig. 5B, step S5 – 7; specification at page 18, lines 15 – 22; and page 57, lines 10 - 15).

According to claim 19, a method for facilitating a windowed content manifestation environment within a web browser comprises the steps of transmitting a software system and associated content (Fig. 3, step S3-2) via an electronic data network (specification at page 13, lines 7 – 18; page 20, lines 26 – 31; page 26, lines 31 – 42; page 31, lines 16 – 20; page 33, lines 24 – 28; and page 56, lines 1 - 6) from a server system (Fig. 1B, 102; specification at page 14, lines 3 – 17); operating a web browser client (Fig. 1C; specification at page 10, lines 14 – 17; Fig. 3, step S3-2) within a data processing system (108) that is coupled to said server system (102) at least in part via said electronic data network (Fig. 1A); initiating a content manifestation environment within said web browser client (Figs. 2A and 2B; specification at page 7, lines 22 – 24; page 18, lines 9 – 11; page 19, lines 12 – 21; and page 20, line 15 *et seq.*); receiving said software system and said associated content via said server system (specification at page 15, lines 15 – 18; page 21, lines 4 – 8; page 30, line 45 – page 31, line 4; Fig. 5A, step S5-5; specification at page 56, lines 26 – 31); and processing said software system and said associated content to produce at least one window object within said content manifestation environment (Fig. 5B, step S5 – 7; specification at page 18, lines 15 – 22; and page 57, lines 10 - 15), said at least one window object associated with a set of controllable attributes and configured to manifest at least a portion of said associated content therein, said controllable attributes configured to affect manifestation of said at least one window object by said web browser client within said content manifestation environment (specification at page 23, lines 7 – 20; page 48, lines 52 - 56).

According to claim 30, a network client configured to operate within a data processing system and to receive content from a remote server system to facilitate a windowed content



manifestation environment therein comprises a content retrieval module configured to receive content from a network server system via an electronic data network (specification at page 15, lines 2 – 11); and a processing engine coupled to said content retrieval module (specification at page 15, lines 11 – 14) configured to instantiate a content manifestation environment within the data processing system environment (Figs. 2A and 2B; specification at page 7, lines 22 – 24; page 18, lines 9 – 11; page 19, lines 12 – 21; and page 20, line 15 *et seq.*) to process said content to produce at least one window object (Figs. 2A and 2B, specification at page 14, lines 22 – 26) within said content manifestation environment (Fig. 1D, Fig. 2A, specification at page 18, lines 8 – 15, Fig. 5A, step S5-6; and page 57 lines 1 -8), said at least one window object associated with a set of controllable attributes (specification at page 23, lines 7 – 21; and page 48, lines 52 – 56) and configured to manifest at least a portion of said associated content therein, said controllable attributes configured to affect manifestation of said at least one window object by said the network client within said content manifestation environment (Fig. 5B, step S5 – 7; specification at page 18, lines 15 – 22; and page 57, lines 10 - 15).

According to claim 40 a software system configured to be downloaded by a network server system (Fig. 1B, 102; specification at page 14, lines 3 – 17) to a client system (Fig. 1C; specification at page 10, lines 14 – 17, page 12, lines 17 - 21) via an electronic data network (specification at page 12, lines 25 – 29; page 13, lines 7 – 18; page 20, lines 26 – 31; page 26, lines 31 – 42; page 31, lines 16 – 20; page 33, lines 24 – 28; and page 56, lines 1 - 6) and to control the operation of the network clients comprises an instruction package (page 13, lines 19 – 29) including instructions be processed by a web browser client running within the client system and to control said web browser client to generate a windowed content a manifestation environment therein and to produce at least one window object (Fig. 1D; Fig. 5A, step S5-6; and page 57, lines 1 - 8) within said content manifestation environment (Fig. 3, steps S3-3 and S3-4), said at least one window object associated with a set of controllable attributes (specification at page 23, lines 7 – 21; and page 48, lines 52 – 56) and configured to manifest content therein (Fig. 2A, specification at page 18, lines 8 - 11), said controllable attributes configured to affect manifestation of said at least one window object by said web browser client within said content manifestation environment (Fig. 5B, step S5 – 7; specification at page 18, lines 15 – 22; and page 57, lines 10 - 15).

## **VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL**

Applicant seeks review of the following grounds of rejection set forth in the Final Office Action having a mailing date of January 8, 2007 (hereinafter the “Office Action”):

Whether claims 1- 47 are properly rejected under 35 U.S.C. § 102(b) as being anticipated by Lemay et al., “Laura Lemay’s Web Workshop JavaScript”, Sams.net, 1996, pp. 10-11, 172-186 (hereinafter “*Lemay*”).

## VII. ARGUMENT

### A. Rejections Under 35 U.S.C. § 102(b) In View of *Lemay et al.*

As more fully set forth below, the rejections by the Examiner in the Office Action are improper because:

1. The prior art fails to teach every element of the claims in the arrangement specified by the claims.
2. A description of tools for creating a wide range of functionalities does not anticipate the claimed combination(s) of such functionalities.
3. The applied prior art is not enabled.
4. The dependent claims include limitations not taught or suggested by the reference cited by the Examiner.

Applicant's invention is directed to providing a windowed content manifestation environment within a world wide web (WWW) browser client. A server system is configured to serve a software system and associated content via an electronic data network such as the Internet and WWW. A web browser is coupled to the server system via the electronic data network to provide the content manifestation environment (e.g., a dynamic screen display). The web browser client receives the software system and the associated content via the server system and processes the software system and the associated content to produce at least one window object within the content manifestation environment. The resultant window object(s) are each associated with a set of controllable attributes and are configured to manifest at least a portion of the associated content. The controllable attributes are configured to affect manifestation of the window object(s) by the web browser client within the content manifestation environment. The window object(s) generated within the content manifestation environment may be updated and loaded with content received via the electronic data network (e.g., Internet) without requiring the content manifestation environment to be refreshed (e.g., without requiring screen refresh operations), and in real-time without requiring user intervention such as via hyper-link traversal.

Appellant addresses each of the grounds of rejection below.

Claims 1-47 stand rejected under 35 U.S.C. § 102(b) as being anticipated by *Lemay*. For convenience of the Board, the portions of *Lemay* relied upon in the rejection of the independent claims include the following language:

#### *Testing a Simple JavaScript Program*

*A JavaScript program can be simple – even as small as a single statement – or much more complicated. Take a look at the simple example in Listing 1.2.*

*Listing 1.2. (ALERT.HTM) A script that displays a message in the text and in an alert box.*

---

```
<HTML><HEAD>
<TITLE>Another JavaScript Test
</HEAD>
<BODY>
<SCRIPT LANGUAGE="JavaScript">
document.write("Hello!");
window.alert("Hello again!");
</SCRIPT>
</BODY>
</HTML>
```

---

*This example displays the "Hello!" message as part of the HTML document. In addition, it uses the alert() statement to display a message to the user in a dialog box.*

*What happens when you load this Web page in Netscape? Let's take a quick look at the inner workings of JavaScript. Once you understand what happens and in what order, you'll find it easy to learn the specifics – such as the JavaScript language itself.*

#### *Receiving the Web Page*

*Let's state with a review of the basics: What happens when you request a Web page from a server? This is an important process to understand. Here are the steps involved:*

- 1. You enter a URL into your browser, or select a bookmark.*
- 2. The browser sends an HTTP request for the URL to the appropriate server (known as a GET request).*
- 3. The server sends back the contents of the Web page at the URL.*

4. *The browser sends additional request for each of the graphics included in the Web page.*
5. *After receiving enough information about the graphics to devote the correct amount of space to them, the browser displays the page.*
6. *The remaining graphics are displayed as they are received.*

### *Processing the Script*

*When JavaScript is involved, the process is slightly different. After the full HTML document is received, as in step 3 in the last section, it is examined for the <SCRIPT> tag. If a script is included, it is processed; however, the processing depends on the type of script:*

- ☐ *If the script is included in the header, it is ignored unless a script later calls it.*
- ☐ *If the script is included directly in the body, its output will be included in the Web page – thus, it can affect the display of the page.*
- ☐ *If the script is an event handler for a specific part of the page, it will be processed only when the event happens.*

*All three of these are useful ways of implementing a JavaScript program; you choose one of these methods depending on your needs. In a complicated JavaScript application, you will probably use all three.*

Lemay at 10 – 11.

### *Advanced Browser Features*

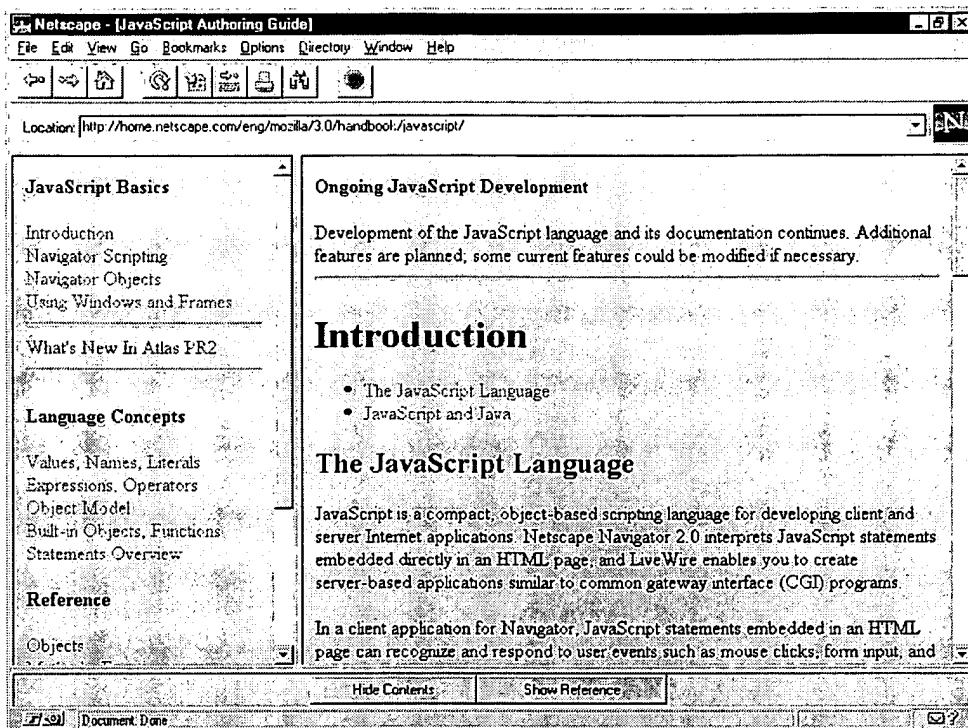
*Before you get into the use of the new features in JavaScript, let's take a quick look at how you use them in a simple HTML page. This will enable you to make them work easily with JavaScript later in this chapter.*

#### *Frames*

*Netscape introduced frames in version 2.0. Recall that frames, also called framesets, are used to divide a Web page's display into multiple sections, each of which can display a different document or portion of a document. Frames are becoming a popular feature in Web pages and are already supported by some non-Netscape browsers.*

*An example of a document that used frames in Netscape's JavaScript Authoring Guide, shown in Figure 9.1. The window is divided into three frames: a frame on the left with the table of contents, one on the right with the actual text, and a small strip on the bottom with two buttons used for*

## JavaScript functions.



Each frame acts as a separate window; it has its own URL. Each frame can also have horizontal and vertical scrollbars. The user can move the dividing lines to resize the different frames. You can enable and disable scrollbars and resizing [sic.] through HTML attributes.

### Defining a Frameset

You use the `<FRAMESET>` tag to define a framed document. This tag is used instead of the `<BODY>` tag. The page with the `<FRAMESET>` declaration defines the layout of the different frames and which documents are loaded into them, but contains no actual data – the contents of each frame are at their own URLs.

The frame definition begins with the `<FRAMESET>` tag and ends with the closing `</FRAMESET>` tag. Within these tags you can't use any ordinary HTML; you can use only tags and attributes that define frames. You can also nest a frameset within another frameset.

The `<FRAMESET>` tag has two attributes: `ROWS` and `COLS`. These define how the window is divided. You can use either of these attributes or both. Both of these attributes have the same value, which is a list of dimensions for each row or column. You can define the rows of columns in the following ways:

- ☐ Specify a numeric value to define the size of a row or column in pixels. For example, "40, 40, 40" defines three frames, each 40 pixels in size. Note that Netscape will expand the frames to fill the

*browser window, though.*

- *Specify a percentage value to allocate a percentage of the window. For example, “20%, 40%, 40%” allocates 20 percent of the window to the first frame and 40 percent to the other two.*
- *Specify a relative value with the asterisk (\*) character. For example, “\*, \*, \*,\*” divides a page evenly into four frames. You can specify a number to give a frame more than one share; for example, “1\*, 2\*” creates two frames, one with 1/3 of the space and the other with 2/3.*

Lemay at 172-173.

It is well settled that to anticipate a claim, the reference must teach every element of the claim. *See* M.P.E.P. § 2131. Moreover, in order for a prior art reference to be anticipatory under 35 U.S.C. § 102 with respect to a claim, “[t]he elements must be arranged as required by the claim.” *See* M.P.E.P. § 2131, citing *In re Bond*, 15 U.S.P.Q.2d 1566 (Fed. Cir. 1990). Furthermore, in order for a prior art reference to be anticipatory under 35 U.S.C. § 102 with respect to a claim, “[t]he identical invention must be shown in as complete detail as is contained in the . . . claim.” *See* M.P.E.P. § 2131, citing *Richardson v. Suzuki Motor Co.*, 9 U.S.P.Q.2d 1913 (Fed. Cir. 1989). Appellant respectfully asserts that the rejection does not satisfy these requirements.

The rejection fails to show that *Lemay* anticipates. The Examiner does not point to a single code listing from *Lemay* that discloses all of the claimed elements of the rejected claims. Rather, the Examiner cites to portions of *Lemay* including generalized capabilities and functionalities that may be implemented using JavaScript. For example, the Examiner refers to Listing 1.2 for describing “a server sending back to a browser, the contents of a web page, which includes a script”. Final Office Action at page 12, lines 15 – 16. The Examiner then skips forward to pages 172 and 173 of the almost 400 page manual for describing what are termed “Advanced Browser Features”. Therein the *Lemay* describes the use of framesets to divide a web page into multiple sections. It is not clear if the Examiner maintains otherwise, but *Lemay* describes dividing a window into frames, not outputting web pages frame *windows*. In addition, having editable tags that may be recoded is not the same as the claimed “set of controllable

attributes.” Even assuming, *arguendo*, that the various code listings in the *Lemay* disclose all of the elements of the claims, they are useless without guidance as to combine them. Thus, distilled to its essence, the rejection assumes that disclosure of all of the claimed elements of an invention is equivalent to disclosure of the invention itself. That is demonstrably false.

The Examiner’s anticipation argument, which relies on the purported disclosures in multiple code listings, is equivalent to handing a box of chemical ingredients to a chemist of ordinary skill and instructing him to synthesize a particular chemotherapy drug. Without providing guidance about which chemicals to use, in what amounts, and how to mix them and in what order, it would be ridiculous to expect the chemist to synthesize the requested drug.

It is further noted that *Lemay* only suggests that certain features are supported by JavaScript but does not provide an enabling disclosure of those features. For example, there is little or no code in the cited portions of *Lemay* implementing the combination of features asserted to be anticipated. For a publication, use, or knowledge to constitute prior art under 35 U.S.C. § 102 (a) or § 102(b), the publication, use or knowledge must satisfy the appropriate statutory temporal limitation and be enabling. See, e.g., *Minnesota Mining & Mfg. Co. v. Chemque, Inc.*, 303 F.3d 1294, 1301 (Fed. Cir. 2002); *In re Paulsen*, 30 F.3d 1475, 1478-79 (Fed. Cir. 1994); *In re LeGrice*, 301 F.2d 929, 933 (C.C.P.A. 1962) (citing treatises).

"In determining that quantum of prior art disclosure which is necessary to declare an applicant's invention 'not novel' or 'anticipated' within section 102, the stated test is whether a reference contains an 'enabling disclosure'... ." *In re Hoeksema*, 399 F.2d 269, 158 USPQ 596 (CCPA 1968). The disclosure in an assertedly anticipating reference must provide an enabling disclosure of the desired subject matter; mere naming or description of the subject matter is insufficient, if it cannot be produced without undue experimentation. *Elan Pharm., Inc. v. Mayo Found. For Med. Educ. & Research*, 346 F.3d 1051, 1054, 68 USPQ2d 1373, 1376 (Fed. Cir. 2003). A reference contains an "enabling disclosure" if the public was in possession of the claimed invention before the date of invention. "Such possession is effected if one of ordinary skill in the art could have combined the publication's description of the invention with his [or her] own knowledge to make the claimed invention." *In re Donohue*, 766 F.2d 531, 226 USPQ 619 (Fed. Cir. 1985). See MPEP §2121.01. *Lemay* only mentions certain capabilities of



JavaScript that are asserted by the Examiner to anticipate the rejected claims. There is no showing that one having ordinary skill in the art would have been able to write the necessary software to implement the functionality asserted to be taught by *Lemay* without requiring extraordinary skill, effort and/or experimentation. To the contrary, Appellant's specification devotes all or most of some 34 pages to presenting approximately 1800 lines of code in support of the pending claims (see Specification at pages 22 – 50 and 52 – 55). In contrast, the cited portion of *Lemay* include scattered segments of code (e.g., Listings 1.2 and 9.1 – 9.8 totaling less than 200 lines).

For the reasons presented it is believed that the outstanding rejection is improper and should be overturned. As required, the rejected claims are specifically addressed below.

1. Claims 1, 14, 19, 30 and 40

In addition to the arguments presented above, each of which is incorporated herein in connection with the following detailed analyses, each of the independent claims are addressed below.

i. *Claim 1*

Claim 1 requires, "... to process said software system and said associated content to produce at least one window object within said content manifestation environment ...." The Examiner alleges that *Lemay* teaches this limitation through its notation of JavaScript's frames capability. January 8, 2007, Office Action, p. 13. It is well-known by those skilled in the art that one of the capabilities of JavaScript is to code or script frames or framesets, which are used to divide a Web page display into multiple sections. However, it is also well-known by those skilled in the art that JavaScript does not require the use of frames or framesets in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of "at least one window object within said content manifestation environment," as required by claim 1. The mere capability to script frames is not the same as producing at least one frame in each script. Thus, Applicant respectfully asserts that, for the

above reasons including those as set forth at pages 13 – 15 herein, claim 1 is patentable over the 35 U.S.C. § 102 rejection of record.

*ii. Claim 14*

Claim 14 requires, “... to process said content to produce at least one window object within said content manifestation environment ....” As noted above, the Examiner alleges that *Lemay*’s notation of JavaScript’s frames capability is the same as this limitation. January 8, 2007, Office Action, p. 15. It is well-known by those skilled in the art that JavaScript does not require the use of frames or framesets in each script written by a developer. Therefore, the use of JavaScript would not result in the production of “at least one window object within said content manifestation environment,” as required by claim 14. Thus, Applicant respectfully asserts that, for the above reasons including those as set forth at pages 13 – 15 herein, claim 14 is patentable over the 35 U.S.C. § 102 rejection of record.

*iii. Claim 19*

Claim 19 requires, “... processing said software system and said associated content to produce at least one window object within said content manifestation environment ...”As noted above, the Examiner alleges that *Lemay*’s notation of JavaScript’s frames capability is the same as this limitation. January 8, 2007, Office Action, p. 16. It is well-known by those skilled in the art that JavaScript does not require the use of frames or framesets in each script written by a developer. Therefore, the use of JavaScript would not result in the production of “at least one window object within said content manifestation environment,” as required by claim 19. Thus, Applicant respectfully asserts that, for the above reasons including those as set forth at pages 13 – 15 herein, claim 19 is patentable over the 35 U.S.C. § 102 rejection of record.

*iv. Claim 30*

Claim 30 requires, “... to process said content to produce at least one window object within said content manifestation environment ....” As noted above, the Examiner alleges that *Lemay*’s notation of JavaScript’s frames capability is the same as this limitation. January 8, 2007, Office Action, p. 16. It is well-known by those skilled in the art that JavaScript does not require the use of frames or framesets in each script written by a developer. Therefore, the use of

JavaScript would not result in the production of “at least one window object within said content manifestation environment,” as required by claim 30. Thus, Applicant respectfully asserts that, for the above reasons including those as set forth at pages 13 – 15 herein, claim 30 is patentable over the 35 U.S.C. § 102 rejection of record.

v. *Claim 40*

Claim 40 requires, “... said web browser client to generate a windowed content manifestation environment therein and to produce at least one window object within said content manifestation environment ....” As noted above, the Examiner alleges that *Lemay*’s notation of JavaScript’s frames capability is the same as this limitation. January 8, 2007, Office Action, p. 16. It is well-known by those skilled in the art that JavaScript does not require the use of frames or framesets in each script written by a developer. Therefore, the use of JavaScript would not result in the production of “at least one window object within said content manifestation environment,” as required by claim 40. Thus, Applicant respectfully asserts that, for the above reasons including those as set forth at pages 13 – 15 herein, claim 40 is patentable over the 35 U.S.C. § 102 rejection of record.

2. *Dependent Claims 2, 20 and 31 are Separately Patentable*

Dependent claims 2, 20 and 31 require that the window object execute within the web browser client which operates within the data processing system. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web browser client. The mere capability to script frames is not the same as producing at least one frame in each script. Thus, Applicant respectfully asserts that, for the above reasons, claims 2, 20 and 31 are patentable over the 35 U.S.C. § 102 rejection of record.

3. Dependent Claims 3, 21 and 32 are Separately Patentable

Dependent claims 3, 21 and 32 require the window object be derived based on instructions processed by the web browser client. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets or window objects derived based on instructions processed by a browser in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object is derived based on instructions processed by the web browser client. The mere capability to script frames is not the same as producing at least one frame in each script. Thus, Applicant respectfully asserts that, for the above reasons, claims 3, 21 and 32 are patentable over the 35 U.S.C. § 102 rejection of record.

4. Dependent Claims 4, 22 and 33 are Separately Patentable

Dependent claims 4, 22 and 33 require that the associated content includes at least one address of a network content source that is configured to download information to the data processing system via the electronic data network, the information to be manifested within the one window within the content manifestation environment. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web browser client. The mere capability to script frames is not the same as producing at least one frame in each script. Thus, Applicant respectfully asserts that, for the above reasons, claims 4, 22 and 33 are patentable over the 35 U.S.C. § 102 rejection of record.

5. Dependent Claims 5, 23 and 34 are Separately Patentable

Dependent claims 5, 23 and 34 require that the associated content includes at least one address of a network content source that is configured to download information to the data processing system via the electronic data network, the information required to be dynamically and continuously manifested within the at least one window object within the content manifestation environment. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets in each script written by a developer or that the information be dynamically and continuously manifested. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web browser client. The mere capability to script frames is not the same as producing at least one frame in each script. Thus, Applicant respectfully asserts that, for the above reasons, claims 5, 23 and 34 are patentable over the 35 U.S.C. § 102 rejection of record.

6. Dependent Claims 6, 24 and 35 are Separately Patentable

Dependent claims 6, 24 and 35 require that the the associated content includes at least one address of a network content source that is configured to download information to the data processing system via the electronic data network, the information to be dynamically and continuously manifested within the at least one window object within the content manifestation environment without requiring the content manifestation environment to be refreshed within the web browser client. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets that are automatically refreshed in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web

browser client and “the information [is] dynamically and continuously manifested ...without requiring the content manifestation environment to be refreshed within the web browser client”. The mere capability to script frames is not the same as producing at least one frame in each script. Thus, Applicant respectfully asserts that, for the above reasons, claims 6, 24 and 35 are patentable over the 35 U.S.C. § 102 rejection of record.

7. Dependent Claims 7, 25 and 36 are Separately Patentable

Dependent claims 7, 25 and 36 require that the controllable attributes associated with the at least one window object permit the window object to be moved within the content manifestation environment. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets including a moveable window in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web browser client including “controllable attributes associated with the at least one window object [so as to] permit the ...window object to be moved within the content manifestation environment”. The mere capability to script frames is not the same as producing at least one frame in each script or providing controllable attributes to permit window object positioning. Thus, Applicant respectfully asserts that, for the above reasons, claims 7, 25 and 36 are patentable over the 35 U.S.C. § 102 rejection of record.

8. Dependent Claims 8, 26 and 37 are Separately Patentable

Dependent claims 8, 26 and 37 require that controllable attributes associated with the window object permit the window object to be resized within the content manifestation environment. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets including resizable windows be present in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the

display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web browser client with “controllable attributes associated with the ...window object [permitting] the ...window object to be resized within the content manifestation environment” . The mere capability to script frames is not the same as producing at least one frame in each script. Thus, Applicant respectfully asserts that, for the above reasons, claims 8, 26 and 37 are patentable over the 35 U.S.C. § 102 rejection of record.

9. Dependent Claims 9, 27 and 38 are Separately Patentable

Dependent claims 9, 27 and 38 require that the controllable attributes associated with the window object permit the window object to be minimized within the content manifestation environment. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets including a minimizable window object within a content manifestation environment in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one minimizable window object within said content manifestation environment,” wherein the window object executes within the web browser client. The mere capability to script frames is not the same as producing at least one frame with a minimizable window object in each script. Thus, Applicant respectfully asserts that, for the above reasons, claims 9, 27 and 38 are patentable over the 35 U.S.C. § 102 rejection of record.

10. Dependent Claims 10, 28 and 39 are Separately Patentable

Dependent claims 10, 28 and 39 require controllable attributes associated with the ... window object that permit the window object to be maximized within the content manifestation environment. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets with the capability to maximize a window object be included in each script written by a developer. In its description

of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web browser client including “controllable attributes associated with the ... window object permit the ... window object to be maximized within the content manifestation environment.” The mere capability to script frames is not the same as producing at least one frame in each script or including specific window functionality. Thus, Applicant respectfully asserts that, for the above reasons, claims 10, 28 and 39 are patentable over the 35 U.S.C. § 102 rejection of record.

11. Dependent Claims 11, 18, 29 and 47 are Separately Patentable

Dependent claims 11, 18, 29 and 47 require that the electronic data network be the Internet. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web browser client and data is transmitted over the Internet. The mere capability to script frames is not the same as producing at least one frame in each script. Thus, Applicant respectfully asserts that, for the above reasons, claims 11, 18, 29 and 47 are patentable over the 35 U.S.C. § 102 rejection of record.

12. Dependent Claim 12 is Separately Patentable

Dependent claim 12 requires that the window object be a tiled window object. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in



general and as disclosed in *Lemay*, would not result in the production of “at least one [tiled] window object within said content manifestation environment,” wherein the [tiled] window object executes within the web browser client. The mere capability to script frames is not the same as producing at least one frame in each script or a tiled window object. Thus, Applicant respectfully asserts that, for the above reasons, claim 12 is patentable over the 35 U.S.C. § 102 rejection of record.

13. Dependent Claim 13 is Separately Patentable

Dependent claim 13 requires that the window object execute within the web browser client which operates within the data processing system and be draggable. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets or draggable window objects be present in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one [draggable] window object within said content manifestation environment,” wherein the [draggable] window object executes within the web browser client. The mere capability to script frames is not the same as producing at least one frame in each script or the provision of draggable window objects. Thus, Applicant respectfully asserts that, for the above reasons, claims 2, 20 and 31 are patentable over the 35 U.S.C. § 102 rejection of record.

14. Dependent Claim 15 is Separately Patentable

Dependent claim 15 requires that the processing engine be further configured to process the content to produce a control section and a content display section within the window object, the content display section configured to at least a portion of the content therein, the control section including a set of controls corresponding to a set of attributes which operate to affect manifestation of the window object and at least a portion of the content within the content display section. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets in each script written by a developer or provision of a control section having a set of attribute controls for affecting

manifestation of the window and content. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web browser client and further including a control section providing for control of manifestation of the window and content. The mere capability to script frames is not the same as producing at least one frame in each script or the provision of the recited controls. Thus, Applicant respectfully asserts that, for the above reasons, claim 15 is patentable over the 35 U.S.C. § 102 rejection of record.

#### 16. Dependent Claim 16 is Separately Patentable

Dependent claim 16 requires that content retrieval module and the processing engine be implemented as sets of computer software objects. Dependent from claim 14, claim 16 necessarily includes the requirements set forth therein including a content retrieval module configured to operate a content manifestation environment to produce a window object manifesting content received by a content retrieval module. While the applied reference provides HTML and JavaScript to provide certain features, JavaScript does not require the use of frames or framesets in each script written by a developer. In its description of the types of scripts that may be processed, *Lemay* only describes one of those types as producing a result that affects the display of a page. *Lemay*, p. 11. Therefore, the use of JavaScript, both in general and as disclosed in *Lemay*, would not result in the production of “at least one window object within said content manifestation environment,” wherein the window object executes within the web browser client. The mere capability to script frames is not the same as producing at least one frame in each script. Thus, Applicant respectfully asserts that, for the above reasons, claim 16 is patentable over the 35 U.S.C. § 102 rejection of record.

#### B. Conclusion

Appellant has provided arguments that overcome the pending anticipatory rejections. The Examiner’s conclusion that the claims should be rejected is unwarranted. Therefore, Appellant respectfully requests that the Board overturn the Examiner’s rejection of claims 1-47.

A copy of the claims involved in the present appeal is attached hereto as Appendix A.

This Appeal Brief is accompanied by payment of the fee for filing a brief in support of an appeal under 41.20(b)(2) together with a petition and corresponding fee for a five month extension of time. If any additional fees are due in connection with this filing, please charge our Deposit Account No. 06-2375, under Order No. P006C1C2C1/10606085 from which the undersigned is authorized to draw and please credit any excess fees to such deposit account.

Dated: January 8, 2008

Respectfully submitted

By

  
Michael J. Strauss

Registration No.: 32,443

FULBRIGHT & JAWORSKI L.L.P.

801 Pennsylvania Avenue, N.W.

Washington, DC 20004-2623

(202) 662-0200

(202) 662-4643 (Fax)

Attorney for Applicant / Appellant

## **APPENDIX A**

### **Claims Involved in the Appeal of Application Serial No. 10/672,945**

1. A system for facilitating a windowed content manifestation environment within a web browser, comprising:

a server system configured to transmit a software system and associated content via an electronic data network; and

a web browser client operating within a data processing system that is coupled to said server system via the electronic data network and having a content manifestation environment, said web browser client operative to receive said software system and said associated content via said server system, to process said software system and said associated content to produce at least one window object within said content manifestation environment, said at least one window object associated with a set of controllable attributes and configured to manifest at least a portion of said associated content therein, said controllable attributes configured to affect manifestation of said at least one window object by said web browser client within said content manifestation environment.

2. The system according to claim 1, wherein said at least one window object executes within said web browser client which operates within said data processing system.

3. The system according to claim 1, wherein said at least one window object is derived based on instructions processed by said a web browser client.

4. The system according to claim 1, wherein said associated content includes at least one address of a network content source that is configured to download information to said data processing system via said electronic data network, said information to be manifested within said at least one window within said content manifestation environment.

5. The system according to claim 1, wherein said associated content includes at least one address of a network content source that is configured to download information to said data processing system via said electronic data network, said information to be dynamically and

continuously manifested within said at least one window object within said content manifestation environment.

6. The system according to claim 1, wherein said associated content includes at least one address of a network content source that is configured to download information to said data processing system via said electronic data network, said information to be dynamically and continuously manifested within said at least one window object within said content manifestation environment without requiring said content manifestation environment to be refreshed within said web browser client.

7. The system according to claim 1, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be moved within said content manifestation environment.

8. The system according to claim 1, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be resized within said content manifestation environment.

9. The system according to claim 1, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be minimized within said content manifestation environment.

10. The system according to claim 1, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be maximized within said content manifestation environment.

11. The system according to claim 1, wherein the electronic data network is the Internet.

12. The system according to claim 1, wherein said at least one window object is a tiled window object.

13. The system according to claim 1, wherein said at least one window object is a draggable window object.

14. A network client configured to operate within a data 2 -processing system and to receive content from a remote server system to facilitate a windowed content manifestation environment, comprising:

a content retrieval module configured to receive content from a network server system via an electronic data network; and

a processing engine coupled to said content retrieval a module configured to operate a content manifestation environment within the data processing system, to process said content to produce at least one window object within said content manifestation environment, said at least one window object 12 configured to manifest at least a portion of said content therein.

15. The network client according to claim 14, wherein said processing engine being further configured to process said content to produce a control section and a content display section within said at least one window object, said content display section configured to at least a portion of said content therein, said control section including a set of controls corresponding to a set of attributes which operate to affect manifestation of said at least one window object and at least a portion of said content within said content display section.

16. The network client according to claim 14, wherein said content retrieval module and said processing engine are implemented as sets of computer software objects.

17. The network client according to claim 14, wherein said content manifestation environment generated by said processing engine is a WWW browser window.

18. The network client according to claim 14, wherein said content retrieval module is configured to receive said content via the Internet.

19. A method for facilitating a windowed content manifestation environment within a web browser, comprising the steps of:

transmitting a software system and associated content via an electronic data network from a server system;

operating a web browser client within a data processing system that is coupled to said server system at least in part via said electronic data network;

initiating a content manifestation environment within said web browser client;

receiving said software system and said associated content via said server system; and processing said software system and said associated content to produce at least one window object within said content manifestation environment, said at least one window object associated with a set of controllable attributes and configured to manifest at least a portion of said associated content therein, said controllable attributes configured to affect manifestation of said at least one window object by said web browser client within said content manifestation environment.

20. The method according to claim 19, wherein said at least one window object executes within said web browser client which operates within said data processing system.

21. The method according to claim 19, wherein said at least one window object is derived based on instructions processed by said web browser client.

22. The method according to claim 19, wherein said associated content includes at least one address of a network content source that is configured to download information to said data processing system via said electronic data network, said information to be manifested within said at least one window object within said content manifestation environment.

23. The method according to claim 19, wherein said associated content includes at least one address of a network content source that is configured to download information to said data processing system via said electronic data network, said information to be dynamically and continuously manifested within said at least one window object within said content manifestation environment.

24. The method according to claim 19, wherein said associated content includes at least one address of a network content source that is configured to download information to said data processing system via said electronic data network, said information to be dynamically and continuously manifested within said at least one window object within said content manifestation environment without requiring said content manifestation environment to be refreshed within said web browser client.

25. The method according to claim 19, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be moved within said content manifestation environment.

26. The method according to claim 19, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be resized within said content manifestation environment.

27. The method according to claim 19, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be minimized within said content manifestation environment.

28. The method according to claim 19, wherein said controllable attributes associated with said at least one window object permitting said at least one window object to be maximized within said content manifestation environment.

29. The method according to claim 19, wherein said electronic data network is the Internet.

30. A network client configured to operate within a data processing system and to receive content from a remote server system to facilitate a windowed content manifestation environment therein, comprising:

a content retrieval module configured to receive content from a network server system via an electronic data network; and

a processing engine coupled to said content retrieval module configured to instantiate a content manifestation environment within the data processing system, to process said content to produce at least one window object within said content manifestation environment, said at least one window object associated with a set of controllable attributes and configured to manifest at least a portion of said associated content therein, said controllable attributes configured to affect manifestation of said at least one window object by said the network client within said content manifestation environment.

31. The network client according to claim 30, wherein said at least one window object executes within the network client.



32. The network client according to claim 30, wherein said at least one window object is derived based on instructions processed by said processing engine.

33. The network client according to claim 30, wherein said associated content includes at least one address of a network a content source that is configured to download information to said data processing system via said electronic data network, said information to be manifested within said at least one window within said content manifestation environment.

34. The network client according to claim 30, wherein said associated content includes at least one address of a network content source that is configured to download information to said data processing system via said electronic data network, said information to be dynamically and continuously manifested within said at least one .window object within said content manifestation environment.

35. The network client according to claim 30, wherein said associated content includes at least one address of a network content source that is configured to download information to said data processing system via said electronic data network, said information to be dynamically and continuously manifested within said at least one window object within said content manifestation environment without requiring said content manifestation environment to be refreshed within the network client.

36. The network client according to claim 30, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be moved within said content manifestation environment.

37. The network client according to claim 30, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be resized within said content manifestation environment.

38. The network client according to claim 30, wherein said controllable attributes associated with said at least one window a object permit said at least one window object to be minimized within said content manifestation environment.

39. The network client according to claim 30, wherein said controllable attributes associated with said at least one window object permit said at least one window object to be maximized within said content manifestation environment.

40. A software system configured to be downloaded by a network server system to a client system via an electronic data network and to control the operation of the network clients, comprising:

an instruction package including instructions be processed by a web browser client running within the client system and to control said web browser client to generate a windowed content a manifestation environment therein and to produce at least one window object within said content manifestation environment, said at least one window object associated with a set of controllable attributes and configured to manifest content therein, said controllable attributes configured to affect manifestation of said at least one window object by said web browser client within said content manifestation environment.

41. The software system according to claim 40, wherein said instruction package controls said web browser client to allow a content stream received by the client system via the electronic data network to be manifested within said at least one window object within said content manifestation environment without causing said content manifestation environment to be refreshed.

42. The software system according to claim 40, wherein said instruction package controls said web browser client to allow a content stream received by said client system via the electronic data network to be manifested within said at least one window object within said content manifestation environment without causing said content manifestation environment to be refreshed.

43. The software system according to claim 40, wherein said content manifestation environment corresponds to a screen environment maintained by said client system.

44. The software system according to claim 40, wherein said instructions are hyper-text mark-up language instructions (HTML).

45. The software system according to claim 40, wherein said instructions are JAVA script instructions.

46. The software system according to claim 40, wherein said instruction package is received by said client system and the web browser client after the web browser client accesses a web site via the electronic data network, said web site serving said instruction package.

47. The software system according to claim 40, wherein said instruction package is intended to be served to said client system via the Internet.

## **APPENDIX B**

### **EVIDENCE**

No evidence pursuant to §§ 1.130, 1.131, or 1.132 or entered by or relied upon by the examiner is being submitted.

**APPENDIX C**

**RELATED PROCEEDINGS**

There being no final decisions rendered by the Court in connection with the proceeding referenced in section II, none are attached.